

Bernd Zwattendorfer, Daniel Slamanig, The Austrian eID ecosystem in the public cloud: How to obtain privacy while preserving practicality, *Journal of Information Security and Applications*, Available online 8 December 2015, ISSN 2214-2126, <http://dx.doi.org/10.1016/j.jisa.2015.11.004>. (<http://www.sciencedirect.com/science/article/pii/S2214212615000642>)

## The Austrian eID Ecosystem in the Public Cloud: How to Obtain Privacy While Preserving Practicality

Bernd Zwattendorfer<sup>a,\*</sup>, Daniel Slamanig<sup>a</sup>

<sup>a</sup>*Graz University of Technology, Inffeldgasse 16a, 8010 Graz, Austria*

---

### Abstract

The Austrian eID system constitutes a main pillar within the Austrian e-Government strategy. The eID system ensures unique identification and secure authentication for citizens protecting access to applications where sensitive and personal data is involved. In particular, the Austrian eID system supports three main use cases: Identification and authentication of Austrian citizens, electronic representation, and foreign citizen authentication at Austrian public sector applications. For supporting all these use cases, several components – either locally deployed in the applications’ domain or centrally deployed – need to communicate with each other. While local deployments have some advantages in terms of scalability, still a central deployment of all involved components would be advantageous, e.g. due to less maintenance efforts. However, a central deployment can easily lead to load bottlenecks because theoretically the whole Austrian population as well as – for foreign citizens – the whole EU population could use the provided services. To mitigate the issue on scalability, in this paper we propose the migration of main components of the ecosystem into a public cloud. However, a move of trusted services into a public cloud brings up new obstacles, particular with respect to privacy. To bypass the issue on privacy, in this paper we propose an approach on how the complete Austrian eID ecosystem can be moved into a public cloud in a privacy-preserving manner by applying selected cryptographic technologies (in particular using proxy

---

\*Corresponding author

Email address: [bernd.zwattendorfer@iaik.tugraz.at](mailto:bernd.zwattendorfer@iaik.tugraz.at) (Bernd Zwattendorfer)

---

re-encryption and redactable signatures). Applying this approach, no sensitive data will be disclosed to a public cloud provider by still supporting all three main eID system use cases. We finally discuss our approach based on selected criteria.

*Keywords:* Electronic identity (eID); identity management; Austrian eID system; cloud computing; public cloud; privacy; proxy re-encryption; redactable signatures.

---

## 1. Introduction

Unique identification and secure authentication are essential processes especially in security-sensitive areas of application such as e-Government or e-Health. In particular, these processes play a key role if sensitive data is processed. To ensure a high level of security for citizen applications in these areas, many European countries have already rolled out national eID solutions supporting unique identification and secure authentication. In Austria, the Austrian citizen card is the official eID for citizens [1].

In general, the Austrian e-Government strategy foresees a thorough eID concept based on the Austrian citizen card, which constitutes the core component for secure identification and authentication of citizens at Austrian e-Government applications. Moreover, the Austrian eID concept also contains representative authentications and authentications of foreign EU citizens, which are treated equally to Austrian citizens in e-Government scenarios. Hence, the main functions of the Austrian eID system are Austrian citizen identification and authentication at online applications, citizen authentication on behalf of a natural or legal person, and the support of foreign citizen authentication at Austrian e-Government applications.

To make these main functions work, the Austrian eID system involves several other components – besides the Austrian citizen card – which are interconnected to each other. Key components, amongst others, are for instance MOA-ID (Module for Online Applications – Identification) [2], an open

source software component locally deployed in each service providers domain facilitating citizen card access, the MIS (Management Issuing Service) [3], which constitutes a central service issuing electronic mandates, or the SPR-GW (SourcePIN Register-Gateway) [4], which a central gateway supporting registration of foreign citizens in Austrian national population registers. Details on the individual components will be given in Section 4.2. Currently, the Austrian eID system treats several deployed MOA-ID instances as well as the MIS and the SPR-GW as trusted entities. While the local deployment model has indeed some benefits, particularly with respect to scalability, in some situations a centralized deployment approach – besides the MIS and the SPR-GW – also of MOA-ID may be preferable. However, in terms of scalability (theoretically the whole Austrian population could use these central services for identification and authentication at service providers) the existing approaches may reach their limits. This can easily lead to load bottlenecks at MOA-ID, the MIS, or the SPR-GW. While the use of electronic mandates and foreign citizen authentications are still in its start-up phase, frequent usages are to be expected in the future. The use of electronic mandates in Austria gets increasing popularity. For instance, professional representation or natural-to-legal person representation constitute daily business in legal procedures. Additionally, representation of parents for their children or children for elderly people are frequent use cases especially in health services. Furthermore, cross-border identifications are steadily increasing because the European Commission currently heavily pushes the STORK framework [5], which will be probably the dominant authentication framework across Europe in the future.

Coping with such increased load may not be easy to handle within the current central deployment scenarios, where each entity is deployed in a trusted data center. Therefore, the authors propose a move of important components of the Austrian eID system (e.g., MOA-ID, MIS, SPR-GW) into a public cloud. Deployment in a public cloud could definitely mitigate any scalability issues due to the characteristics (high scalability, high elasticity, cost reduction, etc.) provided by a public cloud environment. However, a move of such trusted ser-

vice into a public cloud brings up new obstacles, particularly with respect to citizens privacy [6, 7, 8]. Although privacy and security are one of the main issues of public clouds, we still consider the public cloud as the most promising cloud deployment model for a migration of governmental services such as the Austrian eID infrastructure into the cloud. The reasons are – amongst others – particularly the ability to absorb unforeseeable load peaks almost seamlessly and its huge cost savings potential compared to other cloud deployment models [9, 10]. While privacy in the current scenarios is ensured through organizational means, in this paper we illustrate how such a move of trusted services of the Austrian eID system into a public cloud can be successfully realized using cryptographic technologies (by particularly using proxy re-encryption and redactable signatures) by still preserving citizens privacy.

The paper is structured as follows. Section 2 briefly explains related work in the context of identity management. Cryptographic building blocks our work is based on are described in Section 3. In Section 4 the Austrian eID system and its individual components are described in detail. In addition, the three main supported use cases (identification and authentication of Austrian citizens, in representation, and of foreign citizens) and corresponding process flows are explained. How the individual components can be moved into a public cloud in a privacy-preserving manner and how the process flows will change is elaborated in Section 5. In Section 6 we discuss our approach with respect to security, privacy, and practicability. Finally, we draw conclusions in Section 7.

## 2. Related Work

Identity management is no new topic and thus several identity management solutions exist. In this section we briefly outline a couple of identity management systems that have evolved over the past years [11, 12, 13, 14].

First systems arose due to the need of managing employee’s accounts in single organizations. User and identity data was simply stored in directories such as LDAP (Lightweight Directory Access Protocol). In this case, the scope

of the identity management system was limited to this single organization.

Since the need for cross-organizational communication and hence exchanging identification and authentication data across domains gained importance, more sophisticated identity management solutions have established. One early example of such systems is Kerberos [15], which enables secure and uniform authentication in insecure TCP/IP networks. While additionally the WWW became increasingly popular at this time, identity management systems on application level arose.

One example for a central identity management system was Microsoft Passport (latterly called Windows Live ID<sup>1</sup>). Other systems, which follow a decentralized and federated architecture, were the Liberty Alliance Project<sup>2</sup> (that evolved to the Kantara initiative<sup>3</sup>) or Shibboleth<sup>4</sup>. Both projects, Liberty Alliance and Shibboleth, influenced the development of the current version of the Security Assertion Markup Language (SAML 2.0)[16]. SAML defines one of the most important standards dealing with Single Sign-On (SSO) and identity federation at the present time. Contrary to SAML, which is XML-based, OpenID<sup>5</sup> or OpenID Connect<sup>6</sup> rely on more light-weight protocols or data structures for identity data exchange, e.g., simple URL parameters or JSON Tokens. However, both OpenID and OpenID Connect model similar use cases like SAML. Other systems, which are deployed in the field but gained less importance so far are, e.g., WS-Federation[17], Windows CardSpace<sup>7</sup>, or the Central Authentication Service (CAS)<sup>8</sup>.

Unique identification and secure authentication are also essential in sensitive areas of application, e.g., in e-Government or e-Business. Many European

---

<sup>1</sup><https://login.live.com>

<sup>2</sup><http://www.projectliberty.org>

<sup>3</sup><http://kantarainitiative.org>

<sup>4</sup><http://shibboleth.net>

<sup>5</sup>[http://openid.net/specs/openid-authentication-2\\_0.html](http://openid.net/specs/openid-authentication-2_0.html)

<sup>6</sup><http://openid.net/connect/>

<sup>7</sup><http://msdn.microsoft.com/en-us/library/aa480189.aspx>

<sup>8</sup><http://www.jasig.org/cas>

countries have already rolled-out national eID solutions to their citizens, mostly based on smart cards or mobile phones. Examples of such national solutions are the German nPA [18], the Belgian BELPIC [19], or the Austrian citizen card [1] (latter will be detailed in one of the next sections). Most national eID solutions rely on a Public Key Infrastructure (PKI) and X.509 certificates. The Modinis-IDM study [20], the IDABC eID country reports [21], or [22, 23] give an extensive overview of national eID solutions in Europe.

Giving the emerging trend towards cloud computing, identity management gains also importance in this sector. Hence, different cloud identity models have already been defined to cover new requirements particularly relating to cloud computing [24, 25, 26, 27, 28]. The most promising model to fully feature the cloud computing benefits is the operation of an identity provider in the cloud, mostly in the role of an identity broker [29]. Examples for such implementations are Fugen’s Cloud ID Broker<sup>9</sup> or the SkIDentity project<sup>10</sup>. However, those solutions totally neglect any privacy issues with respect to the cloud provider.

To bypass this issue, a handful of privacy-preserving cloud identity management approaches have already emerged in the last years. For instance, Nunez et al. [30] proposed the integration of proxy re-encryption into the OpenID protocol. In follow-up work, they proposed a more generic privacy-preserving cloud identity management model, which they call *BlindIdM* [31]. This model also applies proxy re-encryption but relies on SAML instead of OpenID for the transport protocol. A somewhat related architectural approach – but particularly focusing on eIDs – has been introduced in [32]. A completely different approach based on anonymous credentials for privacy-preservation has been proposed in [33].

Prior to this paper, in [34] we illustrate privacy-preserving design strategies for migrating the basic Austrian eID architecture into the public cloud. The three design strategies proposed there are based on proxy re-encryption, anony-

---

<sup>9</sup><http://fugensolutions.com/cloud-id-broker.html>

<sup>10</sup><http://www.skidentity.com>

mous credentials, and fully homomorphic encryption respectively. Thereby, we conclude that using proxy re-encryption is the most practical approach. However, [34] only investigates the basic use case of the Austrian eID system, namely identification and authentication of Austrian citizens (see Section 4 for details). In this paper we follow a similar approach using proxy re-encryption, but now illustrate the migration of the complete Austrian identity infrastructure into the public cloud. Thereby, we include the two other main uses cases (identification and authentication in representation and foreign citizen authentication), which in part have already been discussed previously in [35, 36]. However, we want to emphasize that it is not a simple combination of these existing results, but we aim at demonstrating that privacy-preserving identity management in public clouds using proxy re-encryption is also possible for complex systems such as the complete Austrian eID ecosystem, which has broad applicability.

### 3. Cryptographic Building Blocks

Subsequently, we review cryptographic building blocks that are required within the proposed approach.

#### 3.1. Digital Signatures

A digital signature scheme (DSS) is a triple  $(\text{DSS.KG}, \text{DSS.Sign}, \text{DSS.Verify})$  of efficient algorithms, where  $\text{DSS.KG}$  is a probabilistic key generation algorithm that takes a security parameter  $\kappa$  and outputs a private and public key pair  $(sk, pk)$ . The (probabilistic) signing algorithm  $\text{DSS.Sign}$  takes as input a message  $m \in \{0, 1\}^*$  and a private (signing) key  $sk$ , and outputs a signature  $\sigma$ . The verification algorithm  $\text{DSS.Verify}$  takes as input a public (verification) key  $pk$ , a message  $m \in \{0, 1\}^*$  and a signature  $\sigma$ , and outputs a single bit  $b \in \{\text{true}, \text{false}\}$  indicating whether  $\sigma$  is a valid signature for  $m$  under  $pk$ . One requires a DSS to be correct, i.e., all honestly generated signatures verify, and existentially unforgeable under adaptively chosen-message attacks (EUF-CMA). In practice one typically employs the hash-then-sign paradigm, i.e., in-

stead of inputting  $m$  into  $\text{DSS.Sign}$  and  $\text{DSS.Verify}$ , one inputs  $H(m)$  where  $H$  is a suitable cryptographic hash function.

### 3.2. (Public Key) Encryption

A public key encryption (PKE) scheme is a triple  $(\text{PKE.KG}, \text{PKE.Enc}, \text{PKE.Dec})$  of efficient algorithms, where  $\text{PKE.KG}$  is a probabilistic key generation algorithm that takes a security parameter  $\kappa$  and outputs a private and public key pair  $(sk, pk)$ . The probabilistic encryption algorithm  $\text{PKE.Enc}$  takes as input a public key  $pk$  and a message  $m \in \{0, 1\}^*$  and returns a ciphertext  $c = \text{PKE.Enc}(pk, m)$ . The decryption algorithm  $\text{PKE.Dec}$  takes as input a private key  $sk$  and a ciphertext  $c$  and returns a message  $m = \text{PKE.Dec}(sk, c)$  or  $\perp$  in the case of failure. A PKE scheme needs to be correct, i.e., decrypting a ciphertext yields the encrypted message, and at least indistinguishable under chosen plaintext attacks (IND-CPA).

Abstractly, we can define private key (or symmetric) encryption schemes (SE) analogously.  $\text{SE.KG}$  generates only a single key  $k$  which is used as input to the encryption and decryption algorithms. For the security of a private key encryption scheme (SE) one also requires at least IND-CPA security. Note that when we speak of applying PKE to a message  $m$ , then we implicitly mean applying *hybrid encryption*, i.e., generating a random key  $k$  of an SE scheme and sending/storing the tuple  $(c_1 = \text{PKE.Enc}(k, pk), c_2 = \text{SE.Enc}(m, k))$ .

### 3.3. Redactable Signatures

A conventional DSS scheme does not allow for alterations of a signed message without invalidating the signature. So called malleable signatures allow to modify (specified) parts of a signed message without invalidating the signature. Malleable signature schemes which allow *removal* of parts (replacement by some special symbol  $\perp$ ) by *any* party are called redactable signature (RS) schemes [37, 38]. Basically, they can be constructed from any secure DSS relying on the hash-then-sign paradigm by virtue of modifying the construction of the hash value (typically using randomized Merkle-Hash trees instead of a plain



cryptographic hash of the entire message). Besides RS constructions for linear documents, there are also approaches for tree-structured documents, e.g., XML documents [39, 40]. Below, we present an abstract definition of redactable signature schemes. Henceforth we assume that a secure scheme for linear documents is used and refer the reader to [37] for required security properties.

**RS.KG:** This probabilistic key generation algorithm takes a security parameter  $\kappa$  and produces and outputs a private and public key pair  $(sk, pk)$ .

**RS.Sign:** This (probabilistic) signing algorithm gets as input the signing key  $sk$  and a message  $m = (m[1], \dots, m[\ell])$ , split into blocks  $m[i] \in \{0, 1\}^*$ , and outputs a signature  $\sigma = \text{RS.Sign}(sk, m)$ .

**RS.Verify:** This deterministic signature verification algorithm gets as input a public key  $pk$ , a message  $m = (m[1], \dots, m[\ell])$ ,  $m[i] \in \{0, 1\}^*$ , and a signature  $\sigma$  and outputs a single bit  $b = \text{RS.Verify}(pk, m, \sigma)$ ,  $b \in \{\text{true}, \text{false}\}$ , indicating whether  $\sigma$  is a valid signature for  $m$  under  $pk$ .

**RS.Redact:** This (probabilistic) redaction algorithm takes as input a message  $m = (m[1], \dots, m[\ell])$ ,  $m[i] \in \{0, 1\}^*$ , a public key  $pk$ , a signature  $\sigma$ , and a list MOD of indices of blocks to be redacted. It returns a modified message and signature pair  $(\hat{m}, \hat{\sigma}) = \text{RS.Redact}(m, pk, \sigma, \text{MOD})$  or an error.

Note that for any redacted signature  $(\hat{m}, \hat{\sigma})$ , we have that  $\text{RS.Verify}(pk, \hat{m}, \hat{\sigma}) = \text{true}$  holds.

### 3.4. Proxy Re-Encryption

Proxy re-encryption (RE) [41] is a public key encryption paradigm where a semi-trusted proxy, given a transformation key, can transform a message encrypted under the key of party  $A$  into another ciphertext to the same message such that another party  $B$  can decrypt with its private key. Although the proxy can perform this re-encryption operation, it does not learn anything about the encrypted message. According to the direction of this re-encryption operation, such schemes can be classified into bidirectional, i.e., the proxy can transform

from  $A$  to  $B$  and vice versa, and unidirectional, i.e., the proxy can convert in one direction only, schemes. Furthermore, one can distinguish between multi-use schemes, i.e., the ciphertext can be transformed from  $A$  to  $B$  to  $C$  etc., and single-use schemes, i.e., the ciphertext can be transformed only once. Moreover, it is desirable that an RE scheme is non-interactive, i.e., a transformation key from  $A$  to  $B$  can be locally computed by  $A$ , where only the public key of  $B$  is required. In this approach we exemplarily use the unidirectional multi-use identity-based proxy re-encryption scheme of Green and Ateniese [42], as in our setting we have a master authority (SRA), which can take care of the key generation. For simplicity we omit the inclusion of the `MaxLevels` parameter (indicating the maximum number of re-encryptions) in our definitions below and note that this parameter needs to be adjusted as required.

**RE.Setup:** This probabilistic algorithm gets a security parameter  $\kappa$ . It outputs the master public parameters  $params$ , which are distributed to users, and the master private key  $msk$ , which is kept private. We assume that  $params$  is available to all algorithms and do not explicitly indicate it.

**RE.KG:** This probabilistic key generation algorithm gets the master private key  $msk$ , and an identity  $id \in \{0,1\}^*$  and outputs a private key  $sk_{id}$  corresponding to identity  $id$ .

**RE.Enc:** This probabilistic encryption algorithm gets an identity  $id \in \{0,1\}^*$ , and a plaintext  $m$  and outputs  $c_{id} = \text{RE.Enc}(id, m)$ .

**RE.RKGen:** This probabilistic re-encryption key generation algorithm gets a private key  $sk_{id_1}$  (derived via RE.KG), and two identities  $(id_1, id_2) \in (\{0,1\}^*)^2$  and outputs a re-encryption key

$$rk_{id_1 \rightarrow id_2} = \text{RE.RKGen}(sk_{id_1}, id_1, id_2).$$

**RE.ReEnc:** This (probabilistic) re-encryption algorithm gets as input a ciphertext  $c_{id_1}$  under identity  $id_1$  and a re-encryption key  $rk_{id_1 \rightarrow id_2}$  (generated

by  $\text{RE.RKGen}$ ) and outputs a re-encrypted ciphertext

$$c_{id_2} = \text{RE.ReEnc}(c_{id_1}, rk_{id_1 \rightarrow id_2}).$$

**RE.Dec:** This decryption algorithm gets a private key  $sk_{id}$ , and a ciphertext  $c_{id}$  and outputs  $m = \text{RE.Dec}(sk_{id}, c_{id})$  or an error  $\perp$ .

We note that as with PKE schemes one requires at least IND-CPA security.

#### 4. The Austrian eID Concept

Unique identification is essential in sensitive areas of applications such as e-Government. Especially if the number of users increases, such as the population of a country, identification based on first name, last name, and date of birth may be ambitious. Therefore, in Austria all citizens are registered in the Central Register of Residence (CRR) and have a unique number assigned (CRR Number). However, this CRR number must not be used directly in e-Government applications due to legal data protection restrictions. Therefore, the SourcePIN Register Authority (SRA), a subdivision of the Austrian Data Protection Commission, encrypts the CRR number to derive a new unique identifier, which is called sourcePIN. The sourcePIN is stored on the Austrian citizen card in conjunction with other identity data such as first name, last name, date of birth, and a qualified signing certificate bound to the citizen's identity. These identification data is wrapped in a special XML-based data structure, the so-called Identity Link. The Identity Link is electronically signed by the SourcePIN Register Authority, which ensures integrity and authenticity of the citizen's identification data on the one side, and, on the other side, certifies the link between identity data and the qualified signing certificate. The Identity Link is finally solely stored on the Austrian citizen card. To provide compact descriptions, we denote the Identity Link in a more general form by  $\mathcal{I} = ((A_1, a_1), \dots, (A_k, a_k))$  as a sequence of attribute labels and attribute values.

To preserve citizen's privacy, it is forbidden by law (based on the Austrian e-Government Act) to directly use the sourcePIN for identification at online

applications. According to this act, the sourcePIN must also never be stored outside the Identity Link. Nevertheless, for still being able to uniquely identify Austrian citizens at applications, the Austrian e-Government concept and strategy foresees a sector-specific identification model. Thereby, a sector-specific PIN (ssPIN) is derived from the combination of the sourcePIN and a governmental sector identifier denoted as  $s$  (e.g., finance, tax, etc.) by using cryptographic one-way hash functions. The use of one-way hash functions still ensures uniqueness of the identifier (ssPIN). In addition, it is not possible to either re-calculate the sourcePIN or an ssPIN of another sector out of a given ssPIN. The ssPIN is finally used for unique identification at online applications.

The entire Austrian eID concept for natural persons relies on citizens being registered in the CRR. However, persons not listed in the CRR (e.g., foreign citizens or Austrian citizens currently residing in a foreign country) can be registered in the so-called Supplementary Register for Natural Persons (SR). By registering in the SR, these persons also get a sourcePIN assigned and hence become part of the Austrian eID infrastructure. This way, foreign citizens can be treated equally to Austrian citizens in online applications. The legal basis for this treatment is the so-called E-Government Equivalence Decree, which became law in 2010.

Besides identification and authentication of natural persons (being either an Austrian or foreign citizen), the Austrian eID concept foresees electronic identification possibilities also for legal persons. Thereby, each legal person can be uniquely identified by a number which has been registered in one of several business registers for legal persons. Such registers are for instance the Company Register or the Central Register for Associations. In general, the overall identification process is based on the usage of electronic mandates. Electronic mandates can be used as electronic representations for legal persons, natural persons, or for professional representatives (e.g., lawyers, notaries, etc.). On a high level, for the representation of legal persons the unique number out of a business register and the representative's sourcePIN (natural person) form the basis information for an electronic mandate. In case of representation of two

natural persons, the sourcePIN of both the representative and the empowering person (mandator) are taken for modelling the electronic mandate.

#### 4.1. *The Austrian Citizen Card Concept*

The Austrian citizen card constitutes the official eID in Austria and is the key component within the Austrian eID concept. The Austrian citizen card concept is rather an abstract definition of a secure eID token than a concrete implementation. Due to this technology-agnostic concept different citizen card implementations exist. Current implementations in use are based on smart cards or mobile phones. However, due to the abstract definition and the technology-neutral concept also alternative approaches and implementations may be developed and rolled-out in the future.

Irrespective of the actual implementation, the citizen card provides the following functionality:

1. Identification and authentication of Austrian citizens
2. Qualified electronic signature creation
3. Encryption and decryption.
4. Data storage

The Austrian citizen card can be used for uniquely *identifying* citizens. Identification is based on the Identity Link, which is solely stored on the citizen card and which includes identifying information such as the sourcePIN. Since the sourcePIN cannot be used directly for identification at online applications because of data protection restrictions, it will be derived according to sectors which results in sector-specific PINs (ssPINs). These ssPINs are finally used for identification at online e-Government applications. *Authentication* by using the Austrian citizen card is carried out by generating a *qualified electronic signature*. The Austrian citizen card is capable for generating qualified electronic signatures according to the EU Signature Directive. Signatures created according to this directive are legally equivalent to handwritten signatures. However,

this functionality is not only used for citizen authentication but also in other applications such as PDF document signing. The third functionality constitutes encryption and decryption. The citizen card includes an additional key pair which allows for secure hardware-based decryption of arbitrary data.<sup>11</sup> Finally, the third citizen card functionality is data storage, where data of arbitrary format (e.g. XML documents or digital certificates) can be stored on the card.

For accessing citizen card functionality irrespective of its implementation, an abstract access layer has been specified. This abstract layer hides implementation specifics from the application and enables access to citizen card functionality by using XML commands. Implementations of this abstract interface are called Citizen Card Software (CCS). The CCS can be either installed locally on the citizen's computer or is provided remotely on a server.

#### 4.2. The Austrian eID Architecture

The overall Austrian eID architecture involves several systems and components. Figure 1 illustrates the Austrian eID architecture separated into operational domains. In the following, we briefly describe the individual components and their basic functionality based on domain separation. Their interactions and individual process flows supporting different use cases will be described in the Section 4.3. Details on the Austrian eID architecture can also be found in [43].

**User Domain:** A *Citizen* wants to access public or private sector service using her Austrian citizen card. The *Citizen Card Software*, which enables easy access to citizen card functionality, usually runs in the citizen's domain.

**Service Provider Domain:** A service provider hosts one or more public or private sector *online applications* providing web-based services to citizens.

---

<sup>11</sup>In the remainder of this paper, we denote the signature key pair of the citizen  $C$  with  $(pk'_C, sk'_C)$  and the encryption key pair with  $(pk_C, sk_C)$ . For details on the formalism, we refer to Section 3.

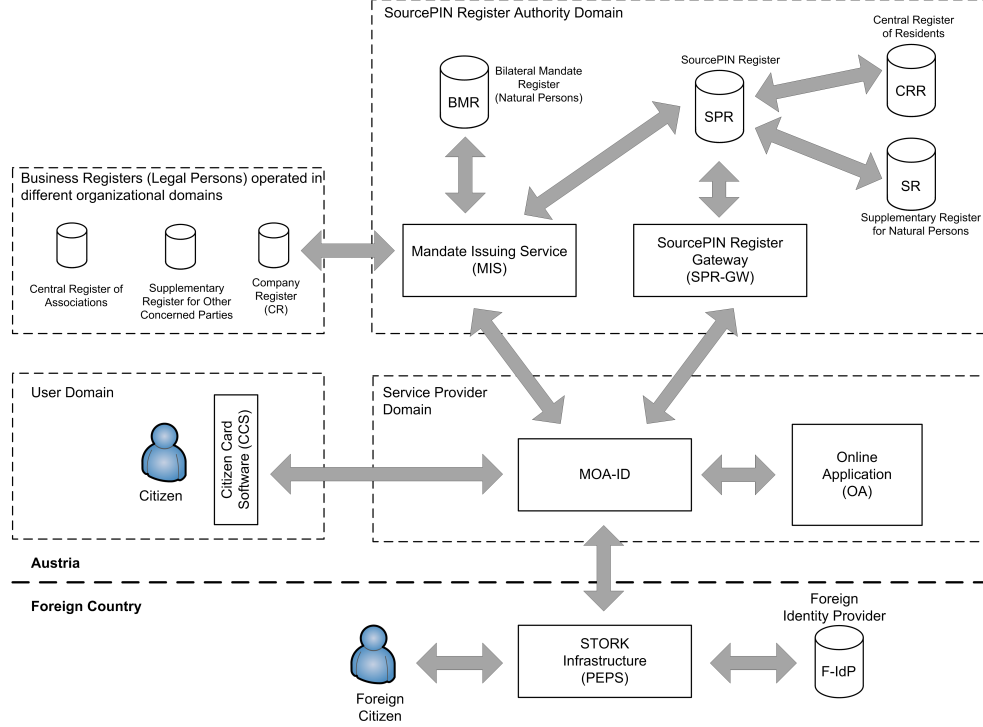


Figure 1: The Austrian eID Architecture

These services require qualified and secure identification and authentication of the Austrian citizen card. Identification and authentication for the application is handled and managed by *MOA-ID*. On the one hand, *MOA-ID* accesses citizen card functionality, and, on the other hand, provides specific and authentic citizen data to the online application for further processing.

**SourcePIN Register Authority Domain:** The *Mandate Issuing Service* (MIS)

is only invoked if citizens authenticate as representative for a natural or legal person. The MIS issues electronic mandates on the fly. For querying appropriate mandate information for natural person representation, the MIS has to query the *Bilateral Mandate Register* (BMR). For fetching appropriate mandate information for representing legal persons, an according

*Business Register* - depending on the type of the legal person - needs to be queried. To finish an authentication process using representation between natural persons, the *SourcePIN Register* (SPR) needs to be queried. The SourcePIN Register is more or less a virtual register, which bundles the information of the *Central Register of Residents* (CRR) and the *Supplementary Register for Natural Persons* (SR). The *SourcePIN Register Gateway*, which is also operated within the SourcePIN Register Authority Domain, is only invoked in the case of foreign citizen authentication. Thereby, the SPR-GW facilitates the registration of foreign citizens in the SR for MOA-ID.

**Business Registers:** In Figure 1 the individual business registers (*Company Register*, *Central Register of Associations*, *Supplementary Register for Other Concerned Parties*) are subsumed under one block for simplicity. However, the individual registers are actually operated in different organizational domains. Operators are for instance the Austrian Ministry of Justice or the Austrian Ministry of the Interior. These registers contain information of legal persons and hence also mandate information for their representation in electronic processes.

**Foreign Country:** In most cases, foreign citizens authenticate via the *STORK infrastructure*. The STORK infrastructure, operated in the foreign country, queries an appropriate *Foreign Identity Provider (F-IdP)* for citizen identification and authentication. Authenticated citizen data are transferred via the STORK infrastructure into the Austrian eID system (more precisely to MOA-ID).

#### 4.3. Identification and Authentication Use Cases

The individual components work all together to support different use cases. In the following we briefly describe three identification and authentication use cases. A detailed description of the interaction and communication between the individual components will be done in Section 5.2.



**Identification and Authentication of Austrian Citizens:** For identification and authentication of Austrian citizens at online applications mainly the component MOA-ID is responsible. MOA-ID handles the identification process by reading and verifying the citizen’s Identity Link, and by deriving the sector-specific PIN from the citizen’s sourcePIN. Authentication is carried out by qualified signature creation, stating the willingness of authenticating at the online application. The citizen’s signature is verified by MOA-ID. The complete identification and authentication process will be illustrated Figure 3, illustrating also the equalities and differences between the current and the cloud process flow. The individual process steps are described in detail in Section 5.2.1.

**Identification and Authentication in Representation:** In addition to MOA-ID, in this scenario the Mandate Issuing Service (MIS) plays an important role. If a citizen wants to represent another person (natural or legal person) in an e-Government application, for successful authentication the citizen needs to provide an authentic electronic mandate to the online application. After the successful identification and authentication of an Austrian citizen, the citizen can select an electronic mandate via the MIS, which empowers her to represent the respective person. Details on this use case will be given in Section 5.2.2. Figure 4 illustrates the identification and authentication scenario when representing a legal person currently and in the proposed cloud-based approach. For simplicity, we limit this use case and its description to legal person representation only, as the representation of natural persons is similar.

**Identification and Authentication of Foreign Citizens:** The Austrian eID concept supports the secure identification and authentication of foreign citizens using their nationally-issued eID. In other words, foreign EU citizens can securely authenticate at Austrian online applications without having the need to apply for an Austrian eID but can use their own national one. For an online application a foreign citizen identification and

authentication process is completely transparent, i.e. the foreign citizen can be treated equally to an Austrian citizen because the same citizen data and data format is used for transmission. For the support of foreign citizen identification and authentication, the Austrian eID architecture relies internally on the SPR-GW and on the external components provided by the foreign country (STORK infrastructure and F-IdP). Details on this use case will be given in Section 5.2.3 and the corresponding Figure 5.

## **5. Porting the Austrian eID Architecture into the Public Cloud**

As can be seen from Figure 1, the individual components have different deployment approaches. For instance, MOA-ID follows a local deployment approach, where each service provider operates one MOA-ID instance in its domain. In comparison to that, the MIS and the SPR-GW are operated centrally in the domain of the SourcePIN Register Authority. Additionally, the deployment of the STORK infrastructure follows a central approach, where each Member State operates a central gateway (PEPS) providing cross-border eID functionality.

While the local deployment of MOA-ID has some clear advantages in terms of end-to-end security or scalability, a central approach may be still advantageous. Citizens could benefit from a central MOA-ID instance as they only need to trust one specific identity provider. Additionally, a central instance of MOA-ID would allow citizens single sign-on across different domains without re-authenticating at each service provider and online application respectively every time. Also service providers can benefit from such an approach as they do not require to run and maintain a separate MOA-ID instance. Naturally, a central deployment approach also has some drawbacks. Namely, a single instance constitutes a single point of failure or attack. Moreover, the level of scalability cannot be reached by a central approach compared to a local or distributed deployment.

Scalability is probably the main issue when considering a central deployment of MOA-ID as all citizen authentication processes will run through this central

instance. This can easily lead to load bottlenecks, as theoretically the whole population of Austria could use this service. The same argument holds for the MIS, the SPR-GW, or the PEPS, which are currently all deployed centrally within a trusted environment. While the use of electronic mandates and cross-border authentications are still in its start-up phase, frequent usages are to be expected in the future.

Dealing with such increased load may not be easy to handle within the current deployment scenarios, where each entity is deployed in a trusted data center. Therefore, we propose a move of the individual entities (MOA-ID, MIS, SPR-GW, PEPS) into a public cloud. Deployment in a public cloud could definitely mitigate any scalability issues due to the characteristics provided by a public cloud environment. However, a move of such trusted services into a public cloud brings up new obstacles, especially with respect to citizen's privacy. While privacy in the current scenarios is ensured through organizational means and mainly relies on trust, in the following sections we illustrate how such a move of trusted services into a public cloud can be successfully realized using cryptographic technologies by still preserving citizens' privacy.

The selected cryptographic technologies for the described approach are based on the results of a previous work [34], which illustrates privacy-preserving design strategies for migrating the basic Austrian eID architecture into the public cloud only. In [34] the only use case that is analyzed is the identification and authentication of Austrian citizens. According to the results in [34], the use of proxy re-encryption and redactable signatures is considered to yield the most practical approach. Thus, in this paper we continue our work by migrating the complete Austrian eID architecture (covering the additional use cases on electronic representation as well as foreign citizen identification and authentication) using proxy re-encryption and redactable signatures. While parts of the two other main use cases (identification and authentication in representation and foreign citizen authentication) have previously been discussed in [35, 36], in this paper we want to show the applicability of the resulting approach of [34] in a complex systems such as the complete Austrian eID ecosystem (with all

required components interacting with each other), which has broad applicability.

### 5.1. The Austrian eID Architecture in the Public Cloud

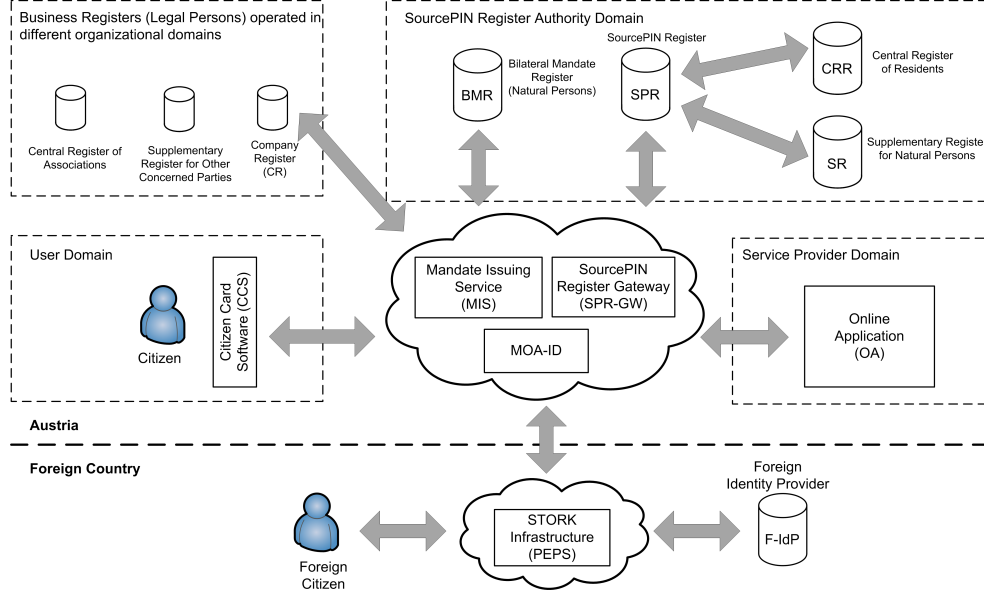


Figure 2: The Austrian eID Architecture in the Public Cloud

Figure 2 illustrates the new architecture of the Austrian eID system when moving important components into the public cloud. In this figure, for simplicity we subsumed the components MOA-ID, MIS, and SPR-GW to be deployed in one public cloud. However, all three components could be operated by different public cloud providers. The STORK PEPS component is assumed to be operated in a different public cloud, as it will be under responsibility of the foreign country.

For being able to move the Austrian eID infrastructure into a public cloud, a few minor changes in the corresponding infrastructure are necessary. In the next sub-section, we explain in detail which changes are required. In the subsequent sections, we describe the adapted process flows of the individual use cases to support an operation of the Austrian eID system in a public cloud.

## 5.2. Identification and Authentication Use Cases

### 5.2.1. Identification and Authentication of Austrian Citizens

Basically, similar to the current situation we assume the SourcePIN Register Authority (SRA) as trusted entity. In this setup scenario, the SRA will be also responsible for the issuance of a slightly modified Identity Link. Additionally, the SRA will manage service provider registration to build appropriate trust relationships between the individual entities.

*Setup.* In the proposed cloud scenario, we assume that the modified Identity Link (denoted by  $\mathcal{I}'$ ) does not contain a sourcePIN but furthermore all ssPINs according to all governmental sectors. Furthermore, all ssPINs are encrypted using a proxy re-encryption scheme, hence every  $(A_i, a_i) \in \mathcal{I}'$  is replaced by the SRA by the encrypted attributes  $c_{a_i} = \text{RE.Enc}(\text{SRA}, a_i)$ . The ssPINs and additional citizen attributes (e.g., name, date of birth) are encrypted under the public key of MOA-ID ( $pk_{\text{MOA-ID}}$ ). The key pair  $(pk_{\text{MOA-ID}}, sk_{\text{MOA-ID}})$  is generated by the SRA. However, the SRA as trusted entity keeps the corresponding private key ( $sk_{\text{MOA-ID}}$ ) and thus MOA-ID will not be able to decrypt the individual attributes. In the current approach, conventional signatures are used to ensure authenticity and integrity of the Identity Link. However, in this cloud-based approach the SRA signs the modified Identity Link using a redactable signature scheme resulting in  $\sigma_{\mathcal{I}'} = \text{RS.Sign}(sk_{\text{SRA}}, \mathcal{I}')$ . By this, each individual attribute of the modified Identity Link can be redacted. The modified Identity Link  $\mathcal{I}'$  is finally stored on the citizen card. In this setup, we further assume that the signature creation certificate stored on the citizen card does not contain any citizen identifying information.

In addition, service providers need to register their online applications at the SRA. We denote the set of service providers  $S = \{S_1, \dots, S_\ell\}$ . For service provider registration, the SRA produces a private key  $sk_{S_j} = \text{RE.KG}(msk_{\text{SRA}}, S_j)$  for  $S_j$  and a re-encryption key  $rk_{\text{MOA-ID} \rightarrow S_j} = \text{RE.RKGen}(sk_{\text{MOA-ID}}, \text{MOA-ID}, S_j)$ . The key  $sk_{S_j}$  is issued to  $S_j$  and  $rk_{\text{MOA-ID} \rightarrow S_j}$  to MOA-ID. We further assume that an appropriate signing key pair  $(pk'_{\text{MOA-ID}}, sk'_{\text{MOA-ID}})$  for MOA-ID is avail-

able.

*Process Flow.* Figure 3 illustrates the process flow combining the current and the cloud-based approach. In fact, the cloud process flow is very similar to the current scenario. However, the differences in the cloud approach compared to the current approach are highlighted in red in Figure 3. In the following, the current eID process flow as well as necessary modifications for the cloud deployment are described in detail.

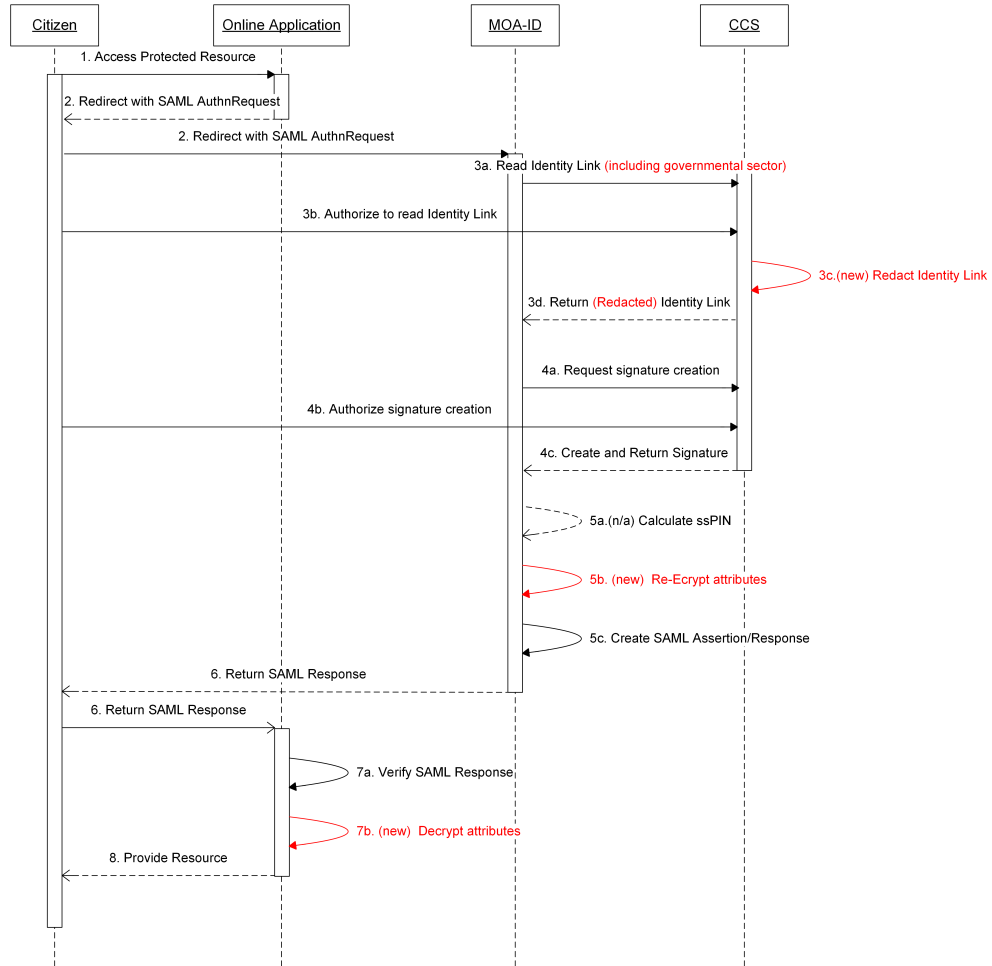


Figure 3: Process flow of Austrian citizen identification and authentication in the cloud approach

1. The citizen wants to access a protected resource at the online application, which requires proper authentication.
2. The online application assembles a SAML authentication request, which is transmitted via HTTP-Redirect to MOA-ID.
- 3a. In this step, MOA-ID sends an appropriate XML request to the CCS for retrieving the Identity Link from the citizen card. For the cloud approach, this request further includes now the governmental sector  $s$ .
- 3b. The citizen authorizes this request appropriately depending on the CCS implementation.
- 3c.(new) By having  $s$ , the CCS can now redact all ssPINs except the ssPIN corresponding to  $s$ .
- 3d. The identity link is returned to MOA-ID and verified. In the cloud approach, the redacted Identity Link  $\mathcal{I}'$  is returned and verified.
- 4a. MOA-ID requests the creation of a qualified electronic signature indicating the willingness of the citizen for online application authentication.
- 4b. The citizen authorizes this request appropriately depending on the CCS implementation.
- 4c. The citizen creates a signature, which is sent back to MOA-ID and verified.
- 5a.(n/a) MOA-ID derives the appropriate ssPIN out of the sourcePIN for the sector the online application belongs to. This step is not applicable in the cloud approach as the sourcePIN as well as the ssPIN are valuable assets which must not be disclosed to the cloud provider.
- 5b.(new) Instead of deriving an ssPIN, MOA-ID re-encrypts the attributes  $c_{a_i}$  of the redacted Identity Link  $\mathcal{I}'$  for the authentication requesting service provider  $S_j$  by using the re-encryption key  $rk_{\text{MOA-ID} \rightarrow S_j}$ . This results in  $c_{S_j} = \text{RE.ReEnc}(rk_{\text{MOA-ID} \rightarrow S_j}, c_{a_i})$ .

- 5c. In the current approach MOA-ID assembles a SAML assertion/response, which includes the ssPIN and additional citizen data out of the Identity Link. In the cloud approach, MOA-ID signs the result coming out with  $\sigma_{\text{MOA-ID}} = \text{DSS.Sign}(sk_{\text{MOA-ID}}, c_{S_j})$ . However, more precisely also in the cloud approach the complete SAML assertion/response is signed.
- 6. MOA-ID returns the SAML assertion/response to the online application via HTTP-POST. Compared to the current approach, where attributes are included in plain in the SAML message, in the cloud approach the SAML message includes re-encrypted attributes only.
- 7a. The online application verifies the SAML response ( $\sigma_{\text{MOA-ID}}$ ), extracts its (encrypted) attributes.
- 7b.(new) The encrypted citizen attributes  $c_{S_j}$  are decrypted using the private key  $sk_{S_j}$ .
- 8. After successful verification, the online application grants access to the resource.

### 5.2.2. Identification and Authentication in Representation

Identification and authentication in representation requires a successful identification and authentication of an Austrian citizen as a prerequisite. After that, the Austrian citizen is eligible to select an electronic mandate containing necessary empowerment information for representing a natural or legal person. In the following, we first give details on the setup for supporting a migration of this use case into a cloud environment. In addition, we give details on the process flow highlight similarities and difference between the current process steps and the steps required in a cloud deployment.

*Setup.* In this scenario, we again assume that the modified Identity Link  $\mathcal{I}'$  is used. Furthermore, in this scenario we additionally rely on the encryption and decryption functionality of the Austrian citizen card. Besides a signature key



pair, each Austrian citizen  $C$  has an encryption key pair  $(pk_C, sk_C)$  stored on her citizen card. This key pair is also generated by the SRA.

In addition to  $(pk_{S_j}, sk_{S_j})$  and  $rk_{MOA-ID \rightarrow S_j}$ , the SRA has to generate additional encryption and re-encryption keys for the individual entities required for mandate processing. For the MIS and for the CR the keys  $(pk_{MIS}, sk_{MIS})$  and  $(pk_{CR}, sk_{CR})$  are created. Since the MIS will be operated in the cloud, the SRA keeps secret  $sk_{MIS}$  and only distributes  $pk_{MIS}$  to the MIS. In addition, the following re-encryption keys are generated:  $rk_{MOA-ID \rightarrow MIS}$ ,  $rk_{MIS \rightarrow CR}$ , and  $rk_{MIS \rightarrow MOA-ID}$ . We further assume that appropriate signing keys are available for the individual entities:  $(pk'_{MOA-ID}, sk'_{MOA-ID})$ ,  $(pk'_{MIS}, sk'_{MIS})$ , and  $(pk'_{CR}, sk'_{CR})$ .

*Process Flow.* Figure 4 illustrates the process flow for representative authentication in the current and in the cloud-based approach. In the following, we describe the process flow in detail.

1. This process step is equal to normal and cloud-based Austrian citizen authentication (cf. Section 5.2.1). However, the citizen indicates that she wants to authenticate on behalf of somebody (e.g., by activating a checkbox).
2. This process step is equal to normal and cloud-based Austrian citizen authentication (cf. Section 5.2.1).
- 3a. In the current scenario, this process step is equal to normal Austrian citizen authentication (cf. Section 5.2.1). For the cloud-based approach, this process step is only similar to cloud-based Austrian citizen authentication. When MOA-ID sends a request for retrieving the Identity Link to the CCS, the request includes the governmental sector the service provider belongs to and the governmental sectors the individual registers storing mandate information belong to.
- 3b.(new) Similar to the cloud-based Austrian citizen authentication approach, the CCS redacts all ssPINs which are not required in this authentication sce-

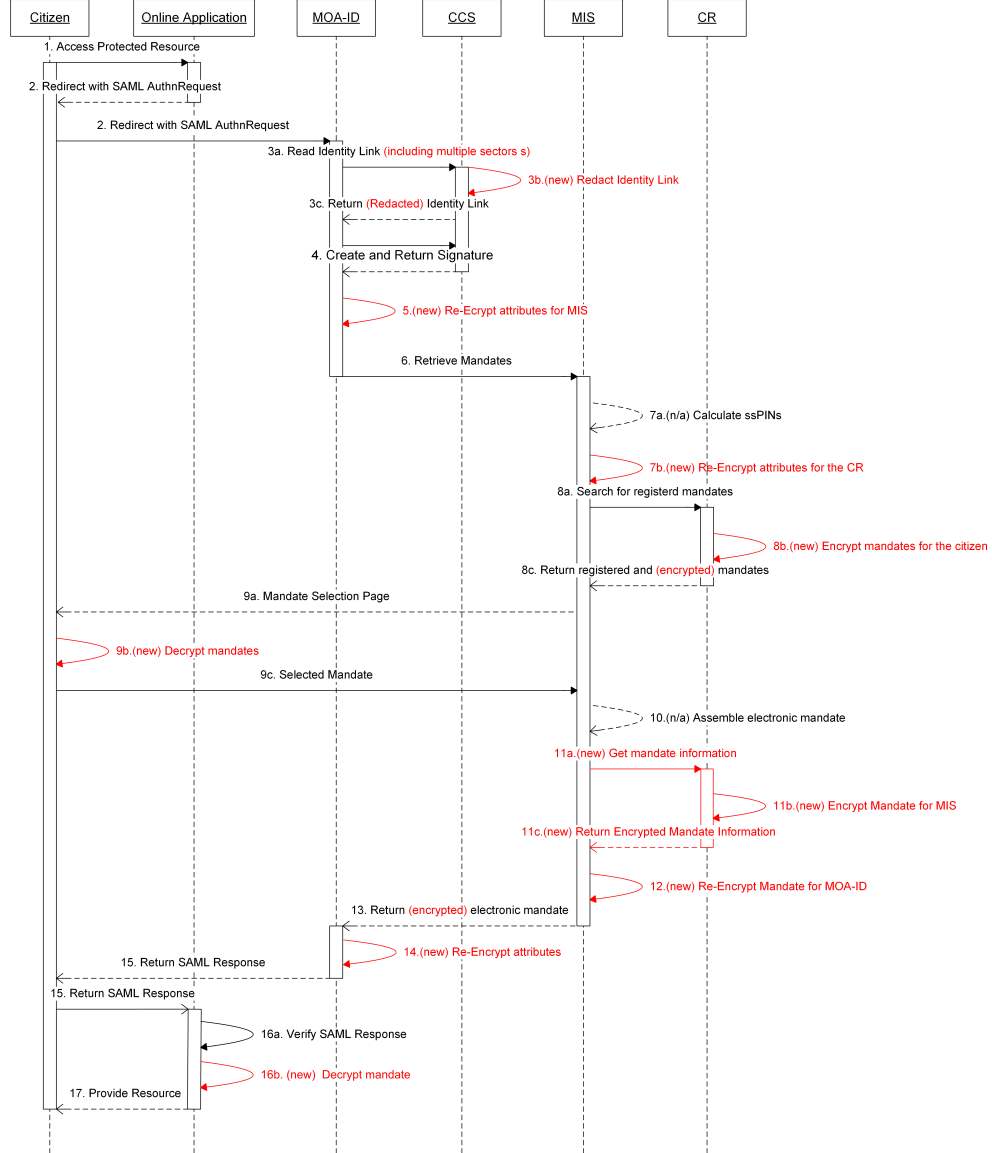


Figure 4: Process flow representing a legal person electronically in the cloud approach

nario. This includes all ssPINs except the one the service provider belongs to ( $ssPIN_{SP}$ ) and the ssPINs required for querying the individual registers for mandate information ( $ssPIN_{CR}$  in this example).

3c. This process step is equal to normal and cloud-based Austrian citizen

authentication (cf. Section 5.2.1).

4. This process step is equal to normal and cloud-based Austrian citizen authentication (cf. Section 5.2.1).
- 5.(new) In this step, MOA-ID re-encrypts the attribute  $ssPIN_{CR}$  from  $\mathcal{I}'$  for the MIS using  $rk_{MOA-ID \rightarrow MIS}$  resulting in  $c_{MIS} = \text{RE.ReEnc}(rk_{MOA-ID \rightarrow MIS}, ssPIN_{CR})$ . This re-encryption result is signed by MOA-ID which outputs  $\sigma_{MOA-ID} = \text{DSS.Sign}(sk'_{MOA-ID}, c_{MIS})$ .
6. Since the citizen wants to authenticate on behalf of somebody, the MIS is queried by MOA-ID for accessing all mandates the citizen is empowered. For that, in the current approach MOA-ID sends the citizen's Identity Link to the MIS.  
  
In the cloud-based approach MOA-ID sends the tuple  $(c_{MIS}, \sigma_{MOA-ID})$  to the MIS for mandate retrieval.
- 7a.(n/a) Out of the sourcePIN from the Identity Link, the MIS calculates all appropriate ssPINs for querying the individual registers. For simplicity, in this scenario the authors illustrate the query process at the company register (CR) only. This step is not applicable in the cloud approach as the sourcePIN as well as the ssPINs are valuable assets which must not to be disclosed to the cloud provider.
- 7b.(new) The MIS verifies  $\sigma_{MOA-ID}$  and re-encrypts  $c_{MIS}$  for the CR using  $rk_{MIS \rightarrow CR}$  and signs the result  $c_{CR}$ . The resulting signature is denoted as  $\sigma_{MIS} = \text{DSS.Sign}(sk'_{MIS}, c_{CR})$ .
- 8a. In the current approach, the MIS searches the CR for registered mandates using the corresponding  $ssPIN_{CR}$  of the citizen. In the cloud-based approach, the MIS sends  $(c_{CR}, \sigma_{MIS})$  to the CR. The CR verifies  $\sigma_{MIS}$ , decrypts  $c_{CR}$ , and searches its register for mandates using the plain  $ssPIN_{CR}$ . In our example, we assume that the mandate information  $mand$  and the corresponding mandate ID  $mandID$  has been found. The

CR signs the mandate and the signature  $\sigma_{CR} = \text{DSS.Sign}(sk_{CR}, mand \| mandID)$  is calculated.

- 8b.(new) Since the citizen is known to the CR (the mandate contains further information of the citizen), it can encrypt the mandate for the citizen using  $pk_C$  resulting in  $c_C = \text{PK.Enc}(pk_C, mand \| mandID \| \sigma_{CR})$ . The CR again signs the encryption result for ensuring integrity and authenticity calculating  $\sigma'_{CR} = \text{DSS.Sign}(sk_{CR}, c_C)$ .
- 8c. In the current approach, the CR returns all registered mandate information for this citizen. In the cloud-based approach, the data  $(c_C, \sigma'_{CR})$  are returned to the MIS, which verifies the signature<sup>12</sup>.
- 9a. The MIS presents the citizen a selection page of all available mandates for her. In the cloud-based example,  $c_C$  is sent to the citizen.
- 9b.(new) The citizen decrypts  $c_C$  and verifies  $\sigma_{CR}$ .
- 9c. The citizen selects the mandate she wants to use for authentication. In this scenario we assume that she wants to act on behalf of a company and thus selects  $mandID$ . In the cloud-based approach, the citizen signs  $mandID$  resulting in  $\sigma_C = \text{DSS.Sign}(sk_C, mandID)$ .  $(mandID, \sigma_C)$  are returned to the MIS.
- 10.(n/a) The MIS assembles all necessary mandate information and signs these data to generate an electronic mandate according to the specification defined by [44]. Amongst others, this electronic mandate contains information of the citizen, who represents the company, the company, and the type of empowerment the citizen is allowed to act on behalf. This step is not applicable in the cloud approach as the mandate information is a valuable asset which must not to be disclosed to the cloud provider.

---

<sup>12</sup>In our scenario, for simplicity the CR has been queried for mandate information only. However, the MIS actually queries all registers that have mandate information available.

- 11a.(new) The MIS queries again the CR for retrieving all necessary information for the selected mandate by using  $(mandID, \sigma_C)$ .
- 11b.(new) The CR calculates  $c_{MIS} = \text{RE.Enc}(MIS, mand || mandID)$  and signs it resulting in  $\sigma''_{CR} = \text{DSS.Sign}(sk'_{CR}, c_{MIS})$ .
- 11c.(new) The CR transmits  $(c_{MIS}, \sigma''_{CR})$  to the MIS.
- 12. The MIS verifies  $\sigma''_{CR}$  and re-encrypts  $c_{MIS}$  for MOA-ID, i.e., computes  $c_{MOA-ID} = \text{RE.ReEnc}(c_{MIS}, rk_{MIS \rightarrow MOA-ID})$ . The MIS signs this re-encryption result  $c_{MOA-ID}$  by calculating the signature  $\sigma'_{MIS} = \text{DSS.Sign}(sk'_{MIS}, c_{MOA-ID})$ . The steps 11a.-12. are only required in the cloud-based approach.
- 13. In the current approach, the MIS returns the electronic mandate to MOA-ID. In the cloud-based approach, the MIS returns  $(c_{MOA-ID}, \sigma'_{MIS})$  to MOA-ID. MOA-ID verifies  $\sigma'_{MIS}$ .
- 14.(new) MOA-ID re-encrypts the data  $c_{MOA-ID}$ , encrypted  $ssPIN_{SP}$ , and  $c_{a_i}$  for  $S_j$  using the key  $rk_{MOA-ID \rightarrow S_j}$ . The result  $c_{S_j}$  is additionally signed using  $sk'_{MOA-ID}$  resulting in  $\sigma'_{MOA-ID}$ . MOA-ID assembles  $(c_{S_j}, \sigma'_{MOA-ID})$  in the SAML response.
- 15. MOA-ID assembles an appropriate SAML assertion/response including the electronic mandate and transmits it to the online application.
- 16a. The online application verifies the response. In the cloud-based approach, the online application verifies the signature  $\sigma'_{MOA-ID}$ .
- 16b.(new) The online application decrypts the mandate and citizen information  $c_{S_j}$  by using the key  $sk_{S_j}$ .
- 17. If verification is successful the online application grants access. The citizen is now able to do online procedures on behalf of the selected company.

### 5.2.3. Identification and Authentication of Foreign Citizens

In this section the identification and authentication of foreign citizen in the current approach and in the cloud approach are described in more detail.

*Setup.* In the previous scenarios we assumed the SRA to be the trusted entity that issues appropriate key material to the involved entities in the Austrian eID system. In this scenario we have to deal with a cross-border scenario, hence we need a trusted entity being able to serve entities across borders. In the current STORK concept, the European Commission (EC) plays a central role managing trust across the involved STORK entities. Therefore, also for our scenario we assume the EC being the entity that issues secure key material to the individual STORK entities and thus we skip a detailed description on that. In this scenario, the EC generates  $(pk_{PEPS}, sk_{PEPS})$  and issues  $pk_{PEPS}$  to the PEPS only. It keeps secret  $sk_{PEPS}$ . Furthermore, the SRA issues  $(pk_{MOA-ID}, sk_{MOA-ID})$ ,  $(pk_{SPR-GW}, sk_{SPR-GW})$ ,  $(pk_{SP}, sk_{SP})$ , and  $(pk_{SR}, sk_{SR})$ . It keeps secret  $sk_{MOA-ID}$  and  $sk_{SPR-GW}$ . The other keys are distributed to the respective entities. In addition, EC generates a re-encryption key  $rk_{PEPS \rightarrow MOA-ID}$  and the SRA the re-encryption keys  $rk_{MOA-ID \rightarrow SPR-GW}$ ,  $rk_{SPR-GW \rightarrow SR}$ , and  $rk_{MOA-ID \rightarrow SP}$ . For further explanations, we denote the identity data of the foreign citizen as  $fc_{data}$ .

We further assume that appropriate signing keys are available for the individual entities:  $(pk'_{MOA-ID}, sk'_{MOA-ID})$ ,  $(pk'_{PEPS}, sk'_{PEPS})$ ,  $(pk'_{F-IDP}, sk'_{F-IDP})$ ,  $(pk'_{SPR-GW}, sk'_{SPR-GW})$ , and  $(pk'_{SR}, sk'_{SR})$ .

*Process Flow.* Figure 5 illustrates the process flow identifying and authenticating a foreign citizen in the current and cloud-based approach. In the following, we describe the process flow in detail.

1. A foreign EU citizen wants to access a service of an Austrian online application.
2. The online application assembles an appropriate SAML authentication request and sends it to MOA-ID.
3. MOA-ID presents the foreign citizen a page where the citizen can select her country of origin.
4. The citizen provides her home country she originates from.

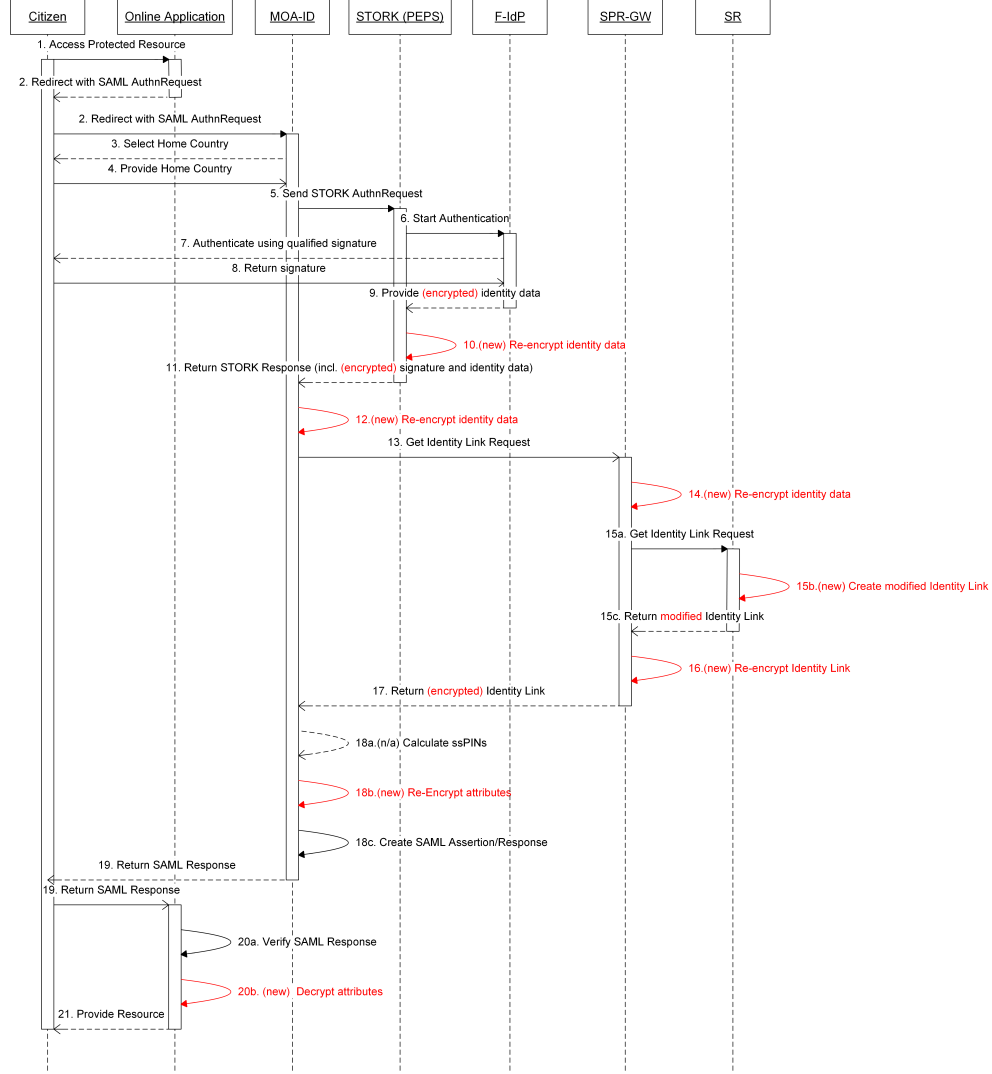


Figure 5: Process flow representing identifying and authenticating a foreign citizen in the cloud approach

5. According to the STORK idea, the foreign citizen will be authenticated in her home country. Therefore, the citizen is redirected to a single gateway (PEPS) in the foreign country, being part of the STORK infrastructure. For starting this authentication process, MOA-ID transmits a STORK authentication request to the foreign PEPS. The PEPS selects an appro-

priate foreign IdP (F-IdP), where the citizen actually authenticates.

6. The PEPS forwards the authentication request to the F-IdP.
7. The F-IdP requests the citizen to authenticate using a qualified signature.
8. The qualified signature is returned to the F-IdP.
9. In the current approach, the F-IdP provides the qualified signature as well as other citizen identifying information (first name, last name, date of birth, identifier) to the PEPS. In the cloud-based approach, we assume the Foreign IdP to be a trusted entity and that it encrypts the foreign citizen's identification data for the PEPS using  $pk_{PEPS}$  resulting in  $c_{PEPS} = \text{RE.Enc}(PEPS, f_{c_{data}})$ . Furthermore,  $c_{PEPS}$  is signed using  $sk'_{F-IdP}$  resulting in  $\sigma_{F-IdP}$ . Both results  $(c_{PEPS}, \sigma_{F-IdP})$  are sent to the PEPS. The PEPS verifies  $\sigma_{F-IdP}$ .
- 10.(new) The PEPS re-encrypts  $c_{PEPS}$  for MOA-ID using  $rk_{PEPS \rightarrow \text{MOA-ID}}$  into  $c_{\text{MOA-ID}}$ . In addition,  $c_{\text{MOA-ID}}$  is signed using  $sk'_{PEPS}$  resulting in  $\sigma_{PEPS}$ .
11. In the current approach, the PEPS assembles the citizen data retrieved from the F-IdP and returns a STORK response to MOA-ID. In the cloud-based approach, the tuple  $(c_{\text{MOA-ID}}, \sigma_{PEPS})$  is sent to MOA-ID. MOA-ID verifies  $\sigma_{PEPS}$ .
- 12.(new) MOA-ID again re-encrypts the foreign citizen data for the SPR-GW to  $c_{\text{SPR-GW}} = \text{RE.ReEnc}(rk_{\text{MOA-ID} \rightarrow \text{SPR-GW}}, c_{\text{MOA-ID}})$ .  $c_{\text{SPR-GW}}$  and the governmental sector  $s$  of the SP is signed resulting in  $\sigma_{\text{MOA-ID}} = \text{DSS.Sign}(sk'_{\text{MOA-ID}}, c_{\text{SPR-GW}} \| s)$ .
13. MOA-ID extracts this information and sends it to the SPR-GW. The SPR-GW verifies the citizen's signature. In the cloud-based approach, the tuple  $(c_{\text{SPR-GW}}, s, \sigma_{\text{MOA-ID}})$  is sent to the SPR-GW. The SPR-GW verifies  $\sigma_{\text{MOA-ID}}$ .
- 14.(new) The SPR-GW does the re-encryption for the SR:  $c_{SR} = \text{RE.ReEnc}(rk_{\text{SPR-GW} \rightarrow \text{SR}}, c_{\text{SPR-GW}})$ . Again, the values  $c_{SR}$  and  $s$  are signed resulting in  $\sigma_{\text{SPR-GW}}$ .



- 15a. The SPR-GW queries the SR to register the foreign citizen in the Supplementary Register for Natural Persons (SR) based on the information received. This registration into the SR is legally based on the Austrian e-Government act [45] and the Austrian e-Government equivalence decree [46]. In the cloud-based approach, the tuple  $(c_{SR}, s, \sigma_{SPR-GW})$  is sent to the SPR-GW.
- 15b.(new) The SR verifies  $\sigma_{SPR-GW}$ , decrypts  $c_{SR}$  using  $sk_{SR}$ , and registers the foreign citizen. During registration, a new modified Identity Link  $\mathcal{I}'$  is created for the foreign citizen. Since the modified Identity Link is created on the fly, it just contains the encrypted  $ssPIN$  for the sector  $s$  and all other  $ssPINs$  are redacted.
- 15c. The SR calculates a sourcePIN for the citizen, creates and assembles an Identity Link, and returns the signed Identity Link to the SPR-GW. In the cloud-based approach, the new modified Identity Link  $\mathcal{I}'$  is encrypted for the SPR-GW using  $pk_{SPR-GW}$  resulting in  $c'_{SPR-GW}$ . The result  $c'_{SPR-GW}$  is signed by applying  $\sigma_{SR} = \text{DSS.Sign}(sk'_{SR}, c'_{SPR-GW})$ . Both results  $(c'_{SPR-GW}, \sigma_{SR})$  are transferred to the SR. The SPR-GW again verifies  $\sigma_{SR}$ .
- 16.(new) The SPR-GW again re-encrypts  $c'_{SPR-GW}$  to  $c'_{MOA-ID} = \text{RE.ReEnc}(rk_{SPR-GW \rightarrow MOA-ID}, c'_{SPR-GW})$ , and signs the result  $c'_{MOA-ID}$  applying  $\sigma'_{SPR-GW} = \text{DSS.Sign}(sk'_{SPR-GW}, c'_{MOA-ID})$ .
17. The SPR-GW returns the Identity Link to MOA-ID. In the cloud-based approach, the re-encrypted Identity Link  $c'_{MOA-ID}$  and  $\sigma'_{SPR-GW}$  are transmitted to MOA-ID. MOA-ID verifies  $\sigma_{SPR-GW}$ .
- 18a.(n/a) MOA-ID derives the appropriate  $ssPIN$  out of the sourcePIN for the sector the online application belongs to. This step is not applicable in the cloud approach as the sourcePIN as well as the  $ssPIN$  are valuable assets which must not to be disclosed to the cloud provider.
- 18b. MOA-ID re-encrypts to  $c_{SP} = \text{RE.ReEnc}(rk_{MOA-ID \rightarrow SP}, c'_{MOA-ID})$ , and signs  $c_{SP}$  applying  $\sigma'_{MOA-ID} = \text{DSS.Sign}(sk'_{MOA-ID}, c_{SP})$ .

- 18c. MOA-ID assembles a SAML assertion/response to be transferred to the SP. In the cloud-based approach the SAML response includes  $c_{SP}$  and  $\sigma'_{\text{MOA-ID}}$ .
- 19.-21. These process steps are equal to normal and cloud-based Austrian citizen authentication (cf. Section 5.2.1). In the cloud-based approach, the online application verifies  $\sigma'_{\text{MOA-ID}}$  and decrypts  $c_{SP}$  using  $sk_{SP}$ .

## 6. Analysis and Discussion of the Proposed Model

In this section we discuss the proposed migration of the Austrian eID system into the public cloud concerning security and privacy as well as practicability aspects.

### 6.1. Security and Privacy Discussion

Our work is based on the assumption that a cloud provider hosting or operating an entity is acting *honest but curious* [47, 31], i.e., the cloud provider operates and works correctly but is not trusted with respect to (data) privacy<sup>13</sup>. In this section we investigate and discuss which personal and sensitive data are disclosed to an entity of the Austrian eID system operated in a public cloud. We thereby compare the information disclosed or seen, respectively, by an entity operated by a public cloud provider. Table 1 illustrates the comparison of the current Austrian eID system and the ported eID system to the cloud with respect to personal or sensitive data disclosed. Since encrypted data is seen as privacy-preserving data, any encrypted data disclosed at an entity in the cloud will not be mentioned.

The comparison Table 1 follows the structure of the previous chapters and sections, where three different use cases for the Austrian eID system are dis-

---

<sup>13</sup>A discussion of security and privacy issues when acting with a totally untrusted cloud provider is out of scope of this work and left for future work. However, under such an assumption data confidentiality and data integrity can still be ensured due to the use of encryption and signature technologies.

tinguished (Identification and Authentication of citizens, in representation, of foreign citizens). In the following, privacy-sensitive data, which is revealed to the individual components, is discussed in detail. However, we only compare those components which are finally ported into the public cloud as they can be considered untrusted with respect to privacy. All other components are trusted and thus do not need a further analysis.

**Identification and Authentication of citizens:**

In this use case only MOA-ID is involved, hence only this component needs to be investigated with respect to privacy. In the current scenario the citizen's identity link (including her name, date of birth and sourcePIN) is exposed to MOA-ID. Additionally, MOA-ID knows the governmental sector of the application the user wants to authenticate and thus also the ssPIN, which is derived out of the citizen's sourcePIN by MOA-ID. Finally, also the citizen's signing certificate is disclosed to MOA-ID.

In contrast to this data set, in the cloud-based approach only the governmental sector of the application the citizen wants to log in remains visible to MOA-ID and the cloud provider respectively. All other data is transferred in encrypted form to MOA-ID only.

**Identification and Authentication in representation:**

In this scenario the components MOA-ID and MIS are involved. Equal of the previous use case, in the current approach MOA-ID gets to know the citizen's identity link (including her name, date of birth and sourcePIN), the citizen's signing certificate, and subsequently the governmental sector of the application and the citizen's ssPIN. The identity link data is also disclosed to the MIS. Since the MIS handles all relevant functionality with respect to authentication on behalf, the MIS also sees all registered mandate information of the citizen. The reason is that the MIS queries all available registers to find existing mandate information for the authenticating citizen. This mandate information is bundled at the MIS for displaying it to the user. After selecting a mandate by the user, the MIS also knows which mandate has been selected. Due to that, the MIS also gets the information of the empowering mandator. Since MOA-ID and

the MIS are interconnected, for fulfilling a successful authentication process in representation all selected mandate information (mandate type, mandator, empowerment, etc.) is also disclosed to MOA-ID. Hence, a lot personal information is disclosed to both components.

Having a look at the cloud approach again, only a minimum of those data are disclosed to MOA-ID and the MIS in this setup. During an authentication process in representation, the MIS only learns the ID (mandID) of the selected mandate by the citizen. In addition, MOA-ID only gets to know the governmental sector of the application. No further privacy-sensitive data is disclosed to either MOA-ID or the MIS because it is processed in encrypted form only.

#### **Identification and Authentication of foreign citizens:**

In this use case the components MOA-ID, SPR-GW, and PEPS are involved. Already at the start of an authentication request, MOA-ID gets to know the citizen's home country because MOA-ID needs to forward the citizen to the respective PEPS for authentication. After successful authentication of the foreign citizen at the F-IdP, all requested citizen data (name, date of birth, unique identifier, signing certificate)<sup>14</sup> is disclosed to the PEPS for further processing. All these data is returned from the PEPS to MOA-ID, thus these data is also disclosed MOA-ID. For registering the foreign citizen in the Austrian Supplementary Register of Residents, these data are forwarded by MOA-ID to the SPR-GW. Out of the signing certificate, the SPR-GW also gets to know the citizen's home country. The SPR-GW registers the citizen and as return data it receives the Austrian identity link of the foreign citizen. This identity link also includes the Austrian unique identifier (sourcePIN), which constitutes sensitive information. The identity link is further exposed to MOA-ID, which uses the governmental sector and the sourcePIN for calculation of the ssPIN.

---

<sup>14</sup>The STORK framework and protocol also supports the transfer of many more attributes. However, for the authentication at an Austrian service provider the name, date of birth, unique identifier, and signing certificate are sufficient, hence we skip a detailed discussion of additional attributes.

Again, compared to all exposed information described above in the current setting, only the governmental sector of the application is disclosed to MOA-ID in the cloud approach. No further sensitive information is disclosed to any of the components MOA-ID, SPR-GW, or PEPS. All data is only available in encrypted form at these components.

## *6.2. Practicability Discussion*

In this section the proposed cloud approach based on selected criteria with respect to to practicability is discussed. The following criteria were selected:

**Re-Use of Existing Infrastructure:** When designing the proposed solution, one criterion was that the existing infrastructure should not severely altered or changed. This particularly includes the architecture and the functionality of the individual components. When comparing the Figures 1 and 2 it can be seen that the overall architecture remains the same instead of the migration of MOA-ID, MIS, SPR-GW, and PEPS into a public cloud. Even the message and transport protocols used to exchange messages between the individual components do not require heavy changes. The exchange messages and protocols need to support the transfer of encrypted data in the cloud-based approach. In addition, also trusted components need to be made capable of encryption functionality. However, this keeps the effort to a minimum, hence no severe changes to the Austrian eID infrastructure need are made.

**Conformance to Current Process Flow:** Another design criterion for our approach was staying conform to the current process flow. While in general conformity is mostly given, changes in the process flow are required to keep the high level of security and privacy-preservation with respect to the cloud providers. However, those process flow changes can be seen as minimal as they are only related to encryption, re-encryption, or additional signature steps.

**Scalability:** The main aim of this work was to guarantee high scalability for central services even at a huge number of users. This aim is mainly realized by migrating important components such as MOA-ID, SPR-GW, or the MIS, where high load can be expected, into public clouds. Of course, load bottlenecks might occur at trusted message endpoints such as individual registers or the F-IdP for the foreign citizen use case. However, those situations are probably very unlikely because not for all citizens the same registers need to be queried at the same time or foreign citizens will not use the same F-IdP at the same time.

**Governance Structure:** For the current situation there is a proper governance structure in place. Meaning, proper trust relationships are (e.g. based on digital certificates) between the individual components. However, the use of re-encryption functionality adds an additional layer of governance requirements. Encryption and decryption key pairs as well as re-encryption keys need to be properly distributed amongst the involved components. This puts some additional complexity to the SRA for use cases involving Austrian citizens only. In addition, also the European Commission and foreign countries are affected as re-encryption functionality is also required across borders. In particular, key management and distribution based on a public key infrastructure (PKI) to the individual components needs to be carried out properly. However, the effort for these tasks can be considered reasonable as the number of involved components is limited (In Austria besides MOA-ID, SPR-GW, and the MIS a few registers and several service providers, in foreign countries besides the PEPS especially different F-IdPs).

### *6.3. Related Work Discussion*

**Related Work on using Proxy Re-Encryption:** One of the first approaches using proxy re-encryption for identity management in the cloud appeared in [30], which integrated proxy re-encryption into the OpenID protocol.

They continued their work by creating a generalized model called *BlindIdM* (A Privacy-Preserving Approach for Identity Management as a Service) [31]. The applicability of *BlindIdM* was further demonstrated by integrating proxy re-encryption functionality into SAML. However, [30, 31] rely on proxy re-encryption for privacy preservation only, whereas our approach additionally uses on redactable signatures, as an additional privacy-preserving mechanism ensuring integrity and authenticity at the same time.

**Related Work on using Anonymous Credentials:** Anonymous Credential systems (aka Privacy ABCs) are a valuable mechanism for ensuring privacy in identity management and have been discussed in context of eID systems, e.g, their integration in the German eID architecture [33]. While early implementations of anonymous credential systems, e.g., idemix [48] on a Java Card [49], however, were too expensive from the user’s (client’s) perspective, state-of-the-art implementations [50, 51, 52] already achieve reasonable efficiency. While anonymous credentials are a valuable means for ensuring privacy in identity management, performance is typically still much slower than when using proxy re-encryption.

## 7. Conclusions

The Austrian eID system plays a major role in the Austrian e-Government strategy. Its main functions are unique identification and secure authentication of Austrian citizens, in representation on behalf of a natural or legal person, or foreign citizen authentication. The current Austrian eID system is based on several components, which are interconnected amongst others. Some of the components are deployed locally (MOA-ID), others centrally (MIS, SPR-GW, PEPS). In general, a central deployment of each individual component is preferable. For instance, a central deployment saves service providers a lot of operational and maintenance costs. However, a central deployment can easily lead to load bottlenecks and scalability issues when the frequency of identification

and authentication processes increases. Theoretically, the entire population of Austria and – going beyond borders – of whole Europe will be able to use and run authentications through the Austrian eID system.

To overcome such scalability bottlenecks in the future, in this paper we presented a solution by moving important centralized services of the Austrian eID system (MOA-ID, MIS, SPR-GW, and PEPS) into a public cloud which considerably improves scalability. By applying appropriate cryptographic technologies we are able to improve scalability by preserving citizen’s privacy with respect to the public cloud providers at the same time. We therefore can conclude that for all identification and authentication use cases no sensitive personal information will be disclosed to a public cloud provider in the cloud-based approach since all data processed in the cloud is encrypted. This strongly preserves citizen’s privacy even if public cloud providers assuming to be acting honest but curious are involved in the proposed architecture. In addition, no major changes to existing infrastructure or the current process flows are required. Only decryption/encryption/re-encryption functionality need to be additionally supported by the individual components and the data transfer protocols must be capable of encrypted data. However, efforts for implementing encryption functionality might be low and encrypted data can also easily be transmitted by standard data exchange protocols such as SAML [31] or OpenID [30]. Finally, a proper governance structure for additional management of encryption/decryption/re-encryption keys needs to be setup. Nevertheless, this can be easily integrated into existing organizational procedures.

## References

## References

- [1] H. Leitold, A. Hollosi, R. Posch, Security Architecture of the Austrian Citizen Card Concept, in: ACSAC, 2002, pp. 391–402.
- [2] T. Lenz, B. Zwattendorfer, K. Stranacher, A. Tauber, Iden-



titätsmanagement in Österreich mit MOA-ID 2.0, eGovernment Review 13 (2014) 20–21.

- [3] H. Leitold, A. Tauber, A systematic approach to legal identity management - best practice austria, in: S. Pohlmann, Reimer (Ed.), ISSE 2011 - Securing Electronic Business Processes, Wiesbaden, 2011, pp. 224 – 234.
- [4] T. Lenz, A modular and flexible attribute mapping service to meet national requirements in cross-border eid federations, in: 13th International Conference on e-Society 2015, 2015, pp. 207 – 214.
- [5] H. Leitold, B. Zwattendorfer, STORK: Architecture, Implementation and Pilots, in: ISSE 2010 Securing Electronic Business Processes, 2011, pp. 1–11.
- [6] S. Pearson, A. Benameur, Privacy, Security and Trust Issues Arising from Cloud Computing, in: IEEE Second International Conference on Cloud Computing Technology and Science, IEEE, 2010, pp. 693–702.
- [7] D. Zissis, D. Lekkas, Addressing cloud computing security issues, Future Generation Computer Systems 28 (3) (2012) 583–592.
- [8] J. Sen, Security and Privacy Issues in Cloud Computing, in: A. R. Martínez, R. Marin-Lopez, F. Pereniguez-Garcia (Eds.), Architectures and Protocols for Secure Information Technology Infrastructures, IGI Global, 2013, pp. 1–45.
- [9] R. Harms, M. Yamartino, THE ECONOMICS OF THE CLOUD FOR THE EU PUBLIC SECTOR, Tech. rep., Microsoft (2010).  
URL [http://www.microsoft.com/global/eu/RichMedia/eu\\_public\\_sector\\_cloud\\_economics\\_a4.pdf](http://www.microsoft.com/global/eu/RichMedia/eu_public_sector_cloud_economics_a4.pdf)
- [10] B. Zwattendorfer, A. Tauber, The public cloud for e-government, International journal of distributed systems and technologies 4 (2013) 1 – 14.

- [11] M. Bauer, M. Meints, M. Hansen, D3.1: Structured Overview on Prototypes and Concepts of Identity Management Systems, Tech. rep., FIDIS (2005).
- [12] M. Dabrowski, P. Pacyna, Overview of Identity Management, Tech. rep., chinacommunications.cn (2008).
- [13] Y. Cao, L. Yang, A survey of Identity Management technology, in: 2010 IEEE International Conference on Information Theory and Information Security, IEEE, 2010, pp. 287–293.
- [14] M. S. Ferdous, R. Poet, A comparative analysis of Identity Management Systems, in: 2012 International Conference on High Performance Computing & Simulation (HPCS), IEEE, 2012, pp. 454–461.
- [15] C. Neuman, T. Yu, S. Hartman, K. Raeburn, RFC 4120: The Kerberos network authentication service (V5), Tech. rep., Network Working Group (2005).
- [16] H. Lockhart, B. Campbell, N. Ragouzis, J. Hughes, R. Philpott, E. Maler, P. Madsen, T. Scavo, Security Assertion Markup Language (SAML) V2.0 Technical Overview, Tech. rep., OASIS (2008).
- [17] M. Goodner, A. Nadalin, Web Services Federation Language (WS-Federation) Version 1.2, Tech. rep., OASIS (2009).
- [18] M. Margraf, The New German ID Card, in: N. Pohlmann, H. Reimer, W. Schneider (Eds.), ISSE 2010 Securing Electronic Business Processes, Vieweg+Teubner, 2010, pp. 367–373.
- [19] D. D. Cock, K. Wouters, B. Preneel, Introduction to the Belgian EID card, in: EuroPKI, Springer-Verlag Berlin Heidelberg, 2004, pp. 1–13.
- [20] D. De Cock, Modinis Overview (2006).  
 URL [http://www.sevecom.org/Presentations/2006-06\\_Paris/Sevecom\\_2006-06-26\\_GModinis-IDM.pdf](http://www.sevecom.org/Presentations/2006-06_Paris/Sevecom_2006-06-26_GModinis-IDM.pdf)

- [21] H. Graux, J. Majava, E. Meyvis, Study on eID Interoperability for PEGS: Update of Country Profiles - Analysis & assessment report, Tech. rep., IDABC (2009).
- [22] S. Arora, Review and Analysis of Current and Future European e-ID Card Schemes, Tech. rep., Royal Holloway, University of London (2008).
- [23] S. Arora, National e-ID card schemes: A European overview, Information Security Technical Report 13 (2) (2008) 46–53.
- [24] A. Gopalakrishnan, Cloud Computing Identity Management, SETLabs Briefings 7 (7) (2009) 45–55.
- [25] J. T. Goulding, identity and access management for the cloud: CA’s strategy and vision, Tech. Rep. May, CA Technologies (2010).  
URL [http://www.ca.com/~media/Files/whitepapers/iam\\_cloud\\_security\\_vision\\_wp\\_236732.pdf](http://www.ca.com/~media/Files/whitepapers/iam_cloud_security_vision_wp_236732.pdf)
- [26] Cloud Security Alliance, SECURITY GUIDANCE FOR CRITICAL AREAS OF FOCUS IN CLOUD COMPUTING V3.0, Tech. rep., Cloud Security Alliance (2011).
- [27] P. Cox, How to Manage Identity in the Public Cloud, InformationWeek reports.  
URL <http://reports.informationweek.com/cart/index/downloadlink/id/8691>
- [28] B. Zwattendorfer, T. Zefferer, K. Stranacher, An Overview of Cloud Identity Management-Models, in: 10th International Conference on Web Information Systems and Technologies (WEBIST), SCITEPRESS Digital Library, 2014, pp. 82–92.
- [29] H. Y. Huang, B. Wang, X. X. Liu, J. M. Xu, Identity Federation Broker for Service Cloud, 2010 International Conference on Service Sciences (2010) 115–120.

- [30] D. Nunez, I. Agudo, J. Lopez, Integrating OpenID with proxy re-encryption to enhance privacy in cloud-based identity services, in: 4th IEEE International Conference on Cloud Computing Technology and Science Proceedings, IEEE, 2012, pp. 241–248.
- [31] D. Nuñez, I. Agudo, BlindIdM: A privacy-preserving approach for identity management as a service, *International Journal of Information Security* (2014) 1–17.
- [32] D. Slamanig, K. Stranacher, B. Zwattendorfer, User-Centric Identity as a Service-Architecture for eIDs with Selective Attribute Disclosure, in: 19th ACM Symposium on Access Control Models and Technologies (SACMAT 2014), ACM, 2014, pp. 153 – 163.
- [33] R. Bjones, I. Krontiris, P. Paillier, K. Rannenberg, Integrating Anonymous Credentials with eIDs for Privacy-respecting Online Authentication, in: *Privacy Technologies and Policy*, 2014, pp. 111–124.
- [34] B. Zwattendorfer, D. Slamanig, Design Strategies for a Privacy-Friendly Austrian eID System in the Public Cloud, *Computers and Security*In press.
- [35] B. Zwattendorfer, D. Slamanig, Privacy-preserving Realization of the STORK Framework in the Public Cloud, in: *SECRYPT 2013 - Proceedings of the 10th International Conference on Security and Cryptography*, 2013, pp. 419–426.
- [36] B. Zwattendorfer, D. Slamanig, Scalable and Privacy-Preserving Variants of the Austrian Electronic Mandate System in the Public Cloud, in: 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, *TrustCom 2013*, 2013, pp. 24–33.
- [37] R. Steinfeld, L. Bull, Y. Zheng, Content Extraction Signatures, in: *ICISC 2001*, Vol. 2288 of LNCS, Springer, 2001, pp. 285–304.
- [38] R. Johnson, D. Molnar, D. X. Song, D. Wagner, Homomorphic Signature Schemes, in: *CT-RSA '02*, Vol. 2271 of LNCS, Springer, 2002, pp. 244–262.

- [39] C. Brzuska, H. Busch, Ö. Dagdelen, M. Fischlin, M. Franz, S. Katzenbeisser, M. Manulis, C. Onete, A. Peter, B. Poettering, D. Schröder, Redactable Signatures for Tree-Structured Data: Definitions and Constructions, in: Applied Cryptography and Network Security, ACNS 2010, Vol. 6123 of LNCS, Springer, 2010, pp. 87–104.
- [40] D. Slamanig, S. Rass, Generalizations and Extensions of Redactable Signatures with Applications to Electronic Healthcare, in: Communications and Multimedia Security, CMS 2010, Vol. 6109 of LNCS, Springer, 2010, pp. 201–213.
- [41] M. Blaze, G. Bleumer, M. Strauss, Divertible Protocols and Atomic Proxy Cryptography, in: Advances in Cryptology - EUROCRYPT '98, Vol. 1403 of LNCS, Springer, 1998, pp. 127–144.
- [42] M. Green, G. Ateniese, Identity-Based Proxy Re-encryption, in: ACNS 2007, Vol. 4521 of LNCS, Springer, 2007, pp. 288–306.
- [43] K. Stranacher, A. Tauber, T. Zefferer, B. Zwattendorfer, The Austrian Identity Ecosystem: An E-Government Experience, in: A. R. Martínez, R. Marin-Lopez, F. Pereniguez-Garcia (Eds.), Architectures and Protocols for Secure Information Technology Infrastructures, IGI Global, 2013, pp. 288–309.
- [44] T. Rössler, A. Hollosi, M. Liehmann, R. Schamberger, Elektronische Vollmachten Spezifikation 1.0.0, Tech. rep., IKT-Strategie des Bundes (2006).  
URL [http://reference.e-government.gv.at/uploads/media/elvm-spez\\_1-0-0\\_20060530.pdf](http://reference.e-government.gv.at/uploads/media/elvm-spez_1-0-0_20060530.pdf)
- [45] Federal Chancellery, The Austrian E-Government Act, Austrian Federal Law Gazette I 7 (2008) 1–11.  
URL <http://www.digitales.oesterreich.gv.at/DocView.axd?CobId=31191>

- [46] Federal Chancellery, Verordnung des Bundeskanzlers, mit der die Voraussetzungen der Gleichwertigkeit gemäß 6 Abs. 5 des E-Government-Gesetzes festgelegt werden (E-Government- Gleichwertigkeitsverordnung) StF: BGBl. II Nr. 170/2010 (2010).  
URL <http://www.ris.bka.gv.at/GeltendeFassung/Bundesnormen/20006801/E-Government-Gleichwertigkeitsverordnung,Fassungvom10.03.2014.pdf>
- [47] Y. Chen, R. Sion, On Securing Untrusted Clouds with Cryptography, in: Proceedings of the 9th annual ACM workshop on Privacy in the electronic society, 2010, pp. 109–114.
- [48] IBM Research, The Identity Mixer (idemix), <http://www.zurich.ibm.com/security/idemix/> (2010).
- [49] P. Bichsel, J. Camenisch, T. Groß, V. Shoup, Anonymous Credentials on a Standard Java Card, in: ACM Conference on Computer and Communications Security, CCS 2009, CCS '09, ACM, New York, NY, USA, 2009, pp. 600–610.
- [50] A. D. L. Piedra, J. Hoepman, P. Vullers, Towards a full-featured implementation of attribute based credentials on smart cards, in: Cryptology and Network Security, CANS 2014, Vol. 8813 of LNCS, Springer, 2014, pp. 270–289. doi:10.1007/978-3-319-12280-9\_18.  
URL [http://dx.doi.org/10.1007/978-3-319-12280-9\\_18](http://dx.doi.org/10.1007/978-3-319-12280-9_18)
- [51] P. Vullers, Efficient Implementations of Attribute-based Credentials on Smart Cards, Ph.D. thesis, Radboud University Nijmegen (2014).
- [52] G. Hinterwälder, F. Riek, C. Paar, Efficient E-cash with Attributes on MULTOS Smartcards, in: Radio Frequency Identification, RFIDsec 2015, Vol. 9440 of LNCS, Springer, 2015, pp. 141–155. doi:10.1007/978-3-319-24837-0\_9.  
URL [http://dx.doi.org/10.1007/978-3-319-24837-0\\_9](http://dx.doi.org/10.1007/978-3-319-24837-0_9)

Table 1: Comparison of personal data disclosure between the current and the cloud-based approach

Approach	Use Case	Component			
		MOA-ID	MIS	SPR-GW	PEPS
Current Approach	Austrian Citizens	<ul style="list-style-type: none"> <li>• Identity Link (name, date of birth, sourcePIN)</li> <li>• ssPIN</li> <li>• Signing certificate</li> <li>• Governmental sector</li> </ul>	-	-	-
	In Representation	<ul style="list-style-type: none"> <li>• Identity Link (name, date of birth, sourcePIN)</li> <li>• ssPIN</li> <li>• Signing certificate</li> <li>• All information of the mandate</li> <li>• Selected mandate for application</li> <li>• Governmental sector</li> </ul>	<ul style="list-style-type: none"> <li>• Identity Link (name, date of birth, sourcePIN)</li> <li>• ssPIN</li> <li>• Signing certificate</li> <li>• All registered mandate information of the citizen</li> <li>• Selected mandate for application</li> <li>• Governmental sector</li> </ul>	-	-
	Foreign Citizens	<ul style="list-style-type: none"> <li>• Citizen's home country</li> <li>• All requested citizen data (name, date of birth, identifier)</li> <li>• Signing certificate</li> <li>• Identity Link (sourcePIN, etc.)</li> <li>• ssPIN</li> <li>• Governmental sector</li> </ul>	-	<ul style="list-style-type: none"> <li>• Citizen's home country</li> <li>• All requested citizen data (name, date of birth, identifier)</li> <li>• Signing certificate</li> <li>• Identity Link (sourcePIN, etc.)</li> </ul>	<ul style="list-style-type: none"> <li>• All requested citizen data (name, date of birth, identifier)</li> <li>• Signing certificate</li> </ul>
Cloud-based Approach	Austrian Citizens	<ul style="list-style-type: none"> <li>• Governmental sector</li> </ul>	-	-	-
	In Representation	<ul style="list-style-type: none"> <li>• Governmental sector</li> </ul>	• MandateID	-	-
	Foreign Citizens	<ul style="list-style-type: none"> <li>• Governmental sector</li> </ul>	-	-	-